Consistency with a set of variables

- **Definition:** An instantiation $(x_{i_1} = d_{i_1}, \ldots, x_{i_u} = d_{i_u})$ is consistent with a set of variables $x_{j_1}, \ldots, x_{j_v}$ if there is an instantiation

  $$(x_{i_1} = d_{i_1}, \ldots, x_{i_u} = d_{i_u}, x_{j_1} = d_{j_1}, \ldots, x_{j_v} = d_{j_v})$$

  that is consistent
  - a consistent solution is consistent with all sets of variables
  - an inconsistent instantiation is inconsistent with all sets of variables

Finding all solutions with CBJ

- When CBJ finds a solution, set the conflict set at level n to be
$$\{nd_1, \ldots, nd_{n-1}\}$$
  - forces chronological backtracking
- Associate a vcf_i (*valid conflict set*) flag at each level
  - clear all flags when a solution is found
  - backtrack chronologically when vcf_i is clear
  - set vcf_i when going forward

BJ and consistency of instantiations

- **Lemma 5:** If BJ performs a backtrack to variable x_h from a deadend at variable x_i, then the instantiation

  $$(x_1 = a_1, \ldots, x_h = a_h)$$

  is inconsistent with x_i

Backtrack rank for CBJ

- Types of CBJ backtracks
  - *A-type* backtracks result from inconsistencies (vcf_i is set)
  - *B-type* backtracks are chronological, caused by searching for additional solutions (vcf_i is clear)
- **Definition:** The *backtrack rank* of an *A-type* backtrack from x_i to x_h is
  - *1* if the backtrack is directly from a dead-end at x_i
  - $d > 1$ if all backtracks performed *to* x_i have rank less than d and at least one has rank $d–1$

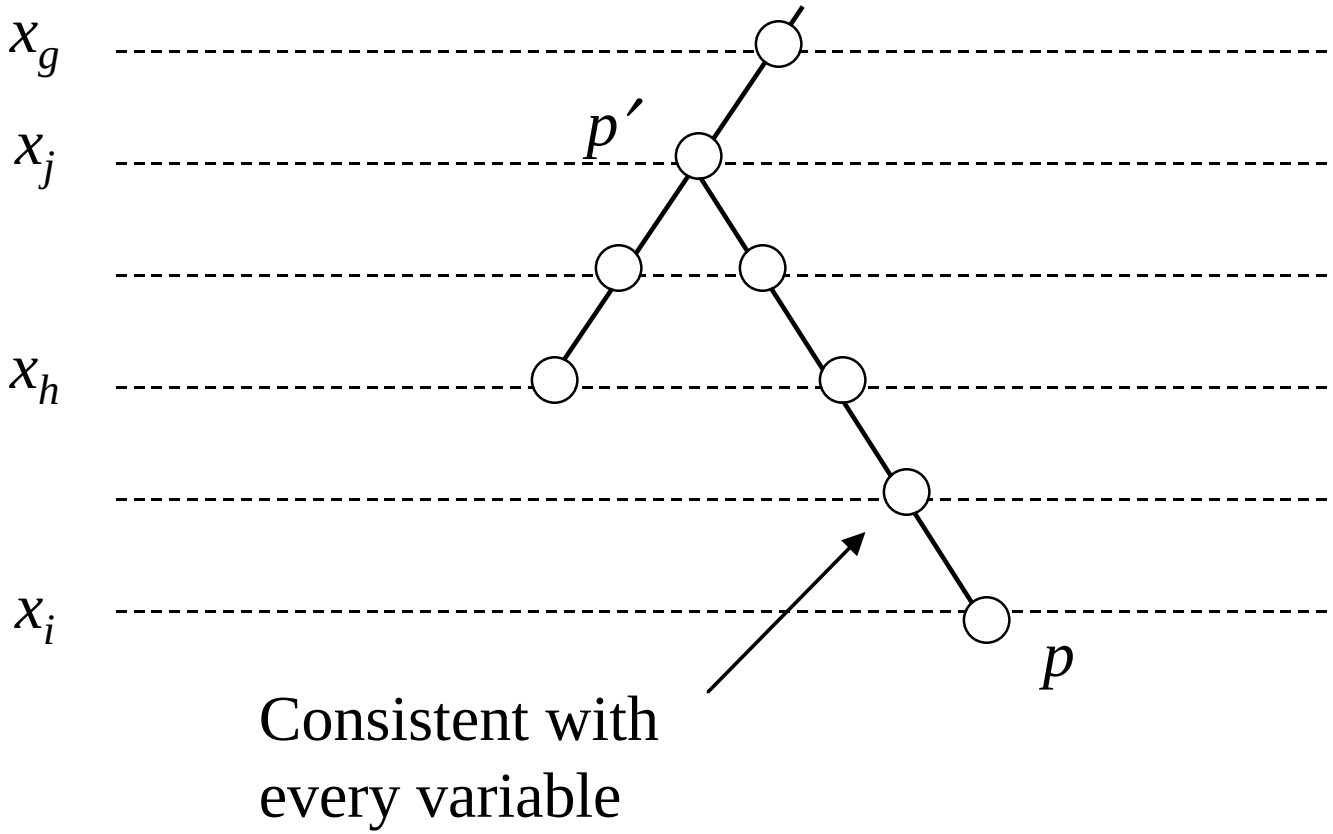
CBJ and consistency of instantiations

- **Lemma 7:** If CBJ performs an *A-type* backtrack from x_i to x_h then there exists a set of variables S such that
  - S is a subset of $\{x_i, \ldots, x_n\}$ and contains x_i; and
  - the instantiation of variables in *conf-set*$_i$ is inconsistent with S

Sufficient conditions to visit a node

- **Theorem 8:**
  - If the parent of a node is consistent, then BT visits the node
  - If the parent of a node is consistent with every variable, then BJ visits the node
  - If the parent of a node is consistent with every set of variables, then CBJ visits the node
  - If a node is consistent and its parent is consistent with every variable, then FC visits the node

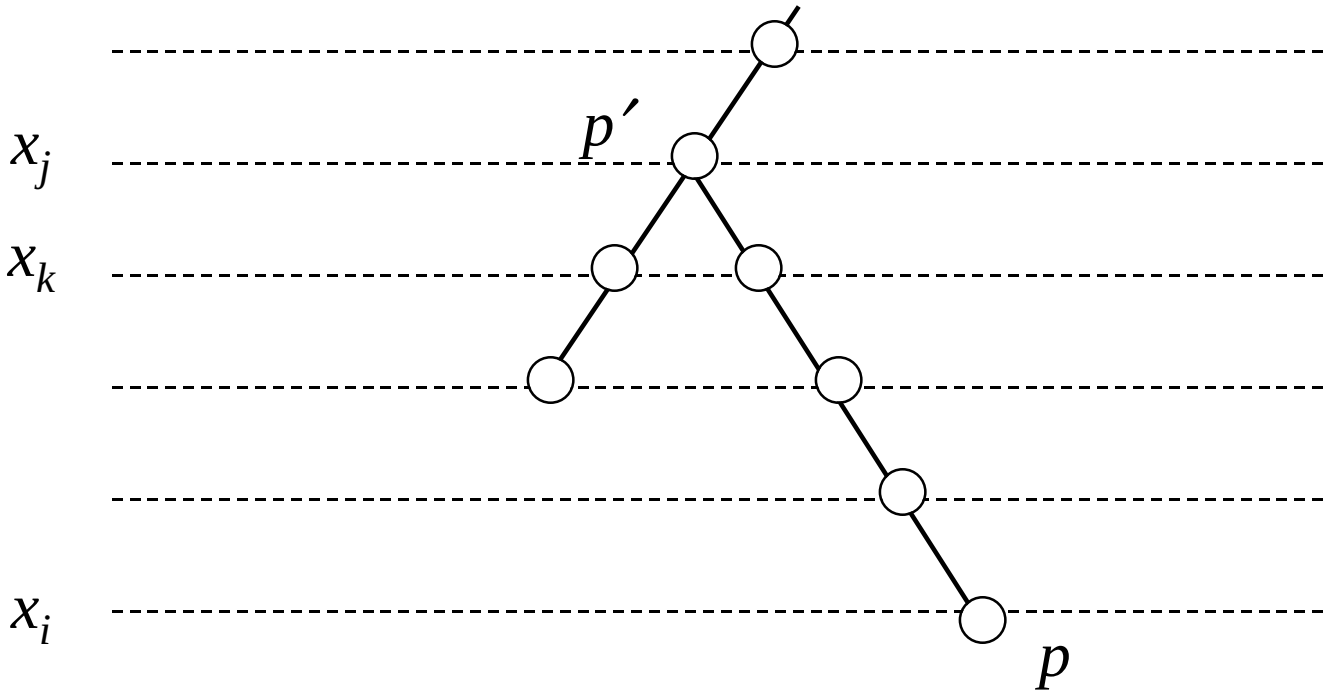
Proof of BJ's sufficient conditions



Necessary conditions to visit a node

- **Theorem 9:**
  - If BT visits a node, then its parent is consistent
  - If BJ visits a node, then its parent is consistent
  - If CBJ visits a node, then its parent is consistent
  - If FC visits a node, then it is consistent and its parent is consistent with every variable

Proof of FC's necessary condition

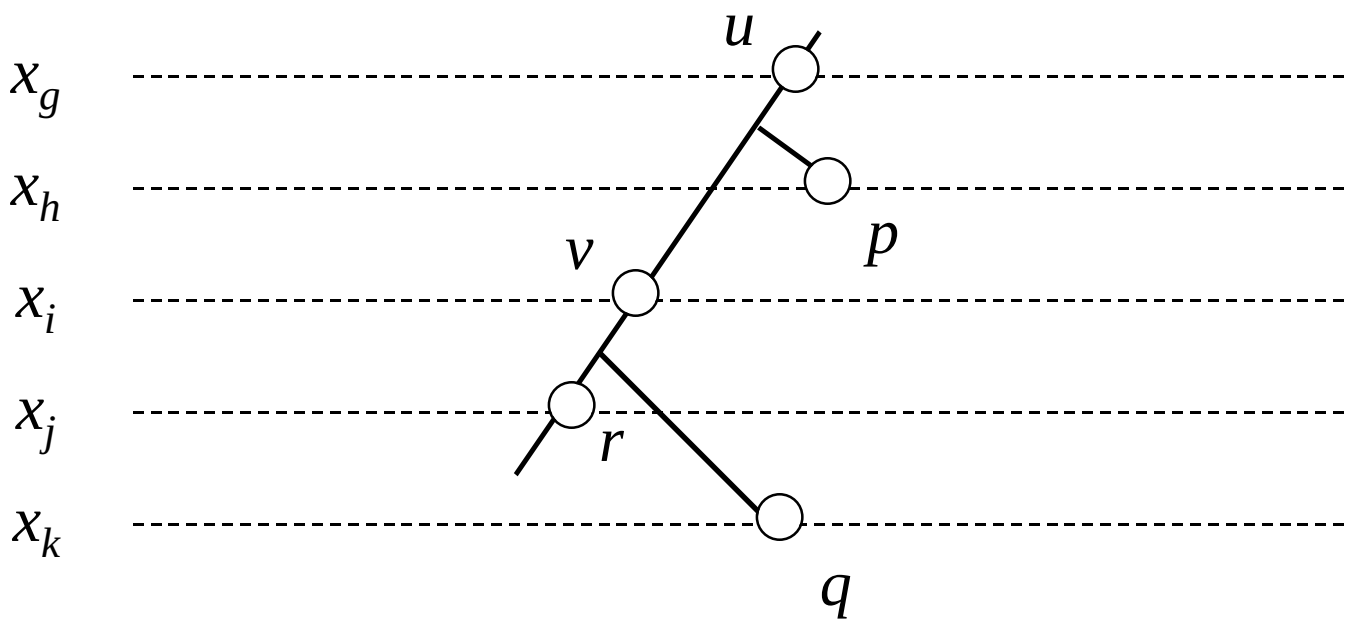


Summary

- **Corollary 10:**
  - BT visits all nodes that BJ visits
  - BT visits all nodes that CBJ visits
  - BT visits all nodes that FC visits
  - BJ visits all nodes that FC visits

Connection between CBJ and BJ

- **Theorem 11:** BJ visits all nodes that CBJ visits



FC-CBJ's necessary condition

- **Theorem 16:** If FC-CBJ visits a node, then it is consistent and its parent is consistent with every variable

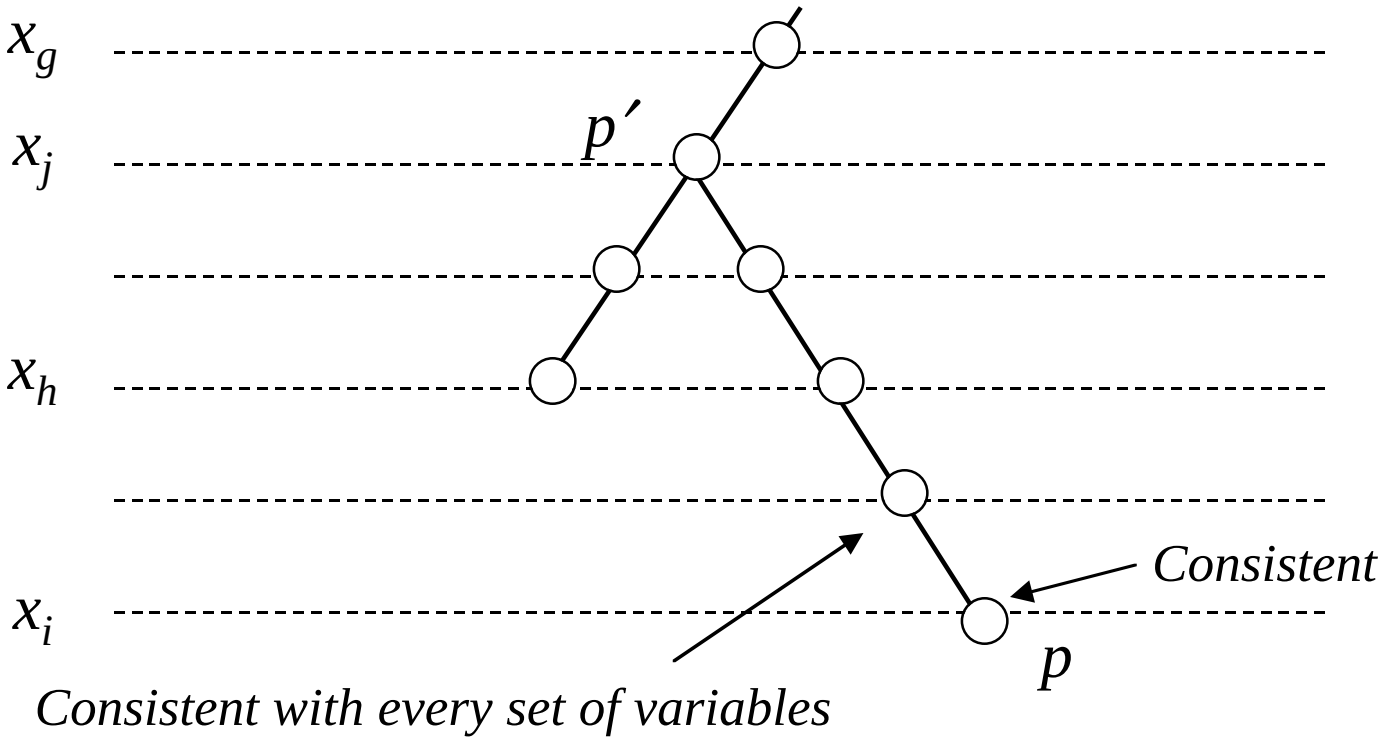
$\Rightarrow$ FC visits every node that FC-CBJ visits

FC-CBJ and consistency of instantiations

- **Lemma 17:** If FC-CBJ performs an *A-type* backtrack from x_i to x_h, then there exists a set of variables S such that
  - S is a subset of $\{x_i, \ldots, x_n\}$ containing x_i; and
  - the instantiation of variables in the conflict set of x_i is inconsistent with S

FC-CBJ's sufficient condition

- **Theorem 18:** If a node is consistent and its parent is consistent with every set of variables, then FC-CBJ visits the node



Correctness of procedures

- **Corollary 12 and 19:**
  - BT is correct
  - BJ is correct
  - CBJ is correct
  - FC is correct
  - FC-CBJ is correct
- In the proofs
  - soundness is established by the necessary condition
  - completeness is established by the sufficient condition

Hierarchy of visited nodes

- Figure 7 from Kondrak and van Beek 1997

Hierarchy of consistency checks

- Figure 8 from Kondrak and van Beek 1997

Single solution and DVO

- Results continue to hold when only single solution are desired
  - one can reformulate all theorems to include an additional condition: "…and the node *precedes* the *termination* node"
- Results continue to hold with DVO provided
  - heuristic for choosing the next variable is deterministic and independent of the backtracking algorithm